



Handling input virtual controlers



INPUT CONTROLERS

Simple Controlers

```
if(Gdx.input.isKeyPressed(Input.Keys.RIGHT){  
    avatar.moveRight();  
}  
else if(Gdx.input.isKeyPressed(Input.Keys.LEFT){  
    ... ..  
}
```

```
if(Gdx.input.isTouched())...
```



PROS FOR CONTROLLERS VIRTUALIZATION

- Easy to change for different controllers
- Modular programming is better
- Abstract code is better for future changes
- Allows multiplexing



EXAMPLES

```
public class VirtualController {  
    public boolean moveRight;  
    public boolean moveLeft;  
}
```

```
public class KeyboardController extends InputAdapter{  
    private VirtualController controller;  
    public KeyboardController(VirtualController controller)...  
  
    //InputAdapters overrides like keyDown must be implemented.  
}
```

```
...//game class{  
    controller = new VirtualController();  
    keyboard = new KeyboardController(controller);  
    Gdx.input.setInputProcessor(keyboard);  
  
    public void render(float delta){  
        if(controller.moveRight) avatar.moveR();  
        ...  
        ...  
    }  
}
```